SYSTEM AND METHODS FOR SYNCHRONIZING INFORMATION AMONG DISPARATE DATASETS

ABSTRACT OF THE DISCLOSURE

A synchronization system with methods for synchronizing information among disparate datasets is described. The system includes the following components. A Source (outbound) "Accessor" is provided for communicating with (i.e., reading from and writing to) a device's data store, such as a source dataset. The Accessor provides open/close and read/write operations on specific dataset types (e.g., Internet Sidekick® address book), provides filtering based on field values, and maintains a Record Map (i.e., the means of identifying records to the system independent of how a record is identified within its own dataset). A corresponding Target Accessor is provided for inbound data, for reading from and writing to the data store of a target device, such as a target dataset. Both Accessors work in conjunction with a "Conduit," which is provided for understanding the data that is being read so that the data can be mapped to a universal schema or "unirecord" (i.e., star topology support) or mapped directly to a target dataset (i.e., mesh topology support). The Conduit serves to broker the services of the Source Accessor and the Target Accessor and to provide field mapping and conversion. Core synchronization functionality is provided by a Synchronizer or "Sync Core" (engine). Its many functions include: initiating synchronization or "SyncSet" (i.e., synchronization data units exchanged between datasets) transfer based on user actions or configured scheduling parameters; applying outbound and inbound synchronization methodologies; brokering the services of a specific Source Accessor (outbound), Conduit, and Transporter; maintaining a Transaction Database; and managing ongoing synchronization operations. The Synchronizer performs its specific functionality on a per record or per data item basis, such as determining which records need to be inserted, deleted, updated, or the like.

JAS:cpu

SF/0014.01.app